Privacy through Pseudonymity in Mobile Telephony Systems

Privacy through Pseudonymity in Mobile Telephony Systems

Myrto Arapinis
University of Edinburgh
marapini@staffmail.ed.ac.uk

Loretta Ilaria Mancini
University of Birmingham
l.mancini@cs.bham.ac.uk

Eike Ritter
University of Birmingham
e.ritter@cs.bham.ac.uk

Mark Ryan
University of Birmingham
m.d.ryan@cs.bham.ac.uk

***Abstract*—To protect mobile phone from tracking by third parties, mobile telephony systems rely on periodically changing pseudonyms. We experimentally and formally analyse the mechanism adopted to update these pseudonyms and point out design and implementation weaknesses that defeat its purpose by allowing the identification and/or tracking of mobile telephony users. In particular, the experiments show that the pseudonym changing mechanism as implemented by real networks does not achieve the intended privacy goals. Moreover, we found out that the standard is flawed and that it is possible to exploit the procedure used to assign a new pseudonym, the TMSI reallocation procedure, in order to track users. We propose countermeasures to tackle the exposed vulnerabilities and formally prove that the 3GPP standard should require the establishment of a fresh ciphering key before each execution of the TMSI reallocation procedure to provide unlinkability.**

I. Introduction

If a third party that eavesdrops on the radio link was able to identify wireless messages as coming from a particular mobile phone, he would be able to track the location of the mobile phone user in real-time. Mobile phone signalling is used for example by market research companies such as [1], [2] in order to track the movements of people within a shopping centre. Contrary to location based service companies, these companies are tracking bearers of mobile phones in an anonymous way yet without their consent, without offering them a service, and sharing the tracking information with parties which have not previously been agreed with the mobile phone bearers. Similar tracking techniques could lead to stalking and other forms of harassment, as well as more mundane invasions of privacy [3]. In order to prevent this, mobile phone protocols employ temporary identifiers (TMSIs) instead of using long-term unique identities (IMSIs) to identify mobile phones. Temporary identities are periodically updated by the network by means of the *TMSI reallocation procedure*. To ensure confidentiality of a newly assigned TMSI, it is transmitted encrypted using a ciphering key.

Our aim in this paper is to analyse what conditions are required in order for this arrangement to guarantee user privacy as intended. In particular, two aspects appear to be important:

1) TMSI reallocation will protect user privacy only if TMSIs are re-allocated often enough, and at the right times (e.g., when users move between locations). The 3GPP standard does not rigorously define the conditions under which TMSI reallocation takes place. We show that the lack of precise directives permits implementations which violate user privacy.
2) The success of TMSI reallocation requires that an attacker with access to the radio channel cannot link the new TMSI to the old one. Encrypting the TMSI in the allocation message is necessary but not sufficient to ensure that. It turns out that other factors, in particular the use of a fresh encryption key for each TMSI reallocation, are also necessary to guarantee unlinkability of old and new TMSIs. The 3GPP standard does not mandate this, again leaving user privacy subject to choices made by network operators.

We analyse the TMSI reallocation procedure from both a *formal* and an *experimental* point of view. Our experimental analysis exposes the adoption by deployed network implementations of weak policies with respect to privacy and hence are vulnerable to tracking mobile phone users. We show that the TMSI reallocation procedure does not provide unlinkability on most of the analysed mobile networks, because:

1) pseudonyms are not updated frequently;
2) the frequency of updates of pseudonyms does not depend on the amount of activity exposing them to tracking adversaries;
3) the same pseudonyms are maintained across different areas, making users linkable within wide areas;
4) it is possible to mount a replay attack on the TMSI reallocation procedure.

All these issues defeat the objective of introducing TMSIs. Our formal analysis allows us to prove the condition under which the TMSI reallocation procedure provides unlinkability. In particular, we formally prove that the establishment of a new encryption key before each execution of the procedure should be a mandatory requirement in the standard specification.

Our Contributions. We present a formal and an experimental analysis of the subscriber's privacy in cellular networks

NDSS '14, 23-26 February 2014, San Diego, CA, USA
 ISBN 1-891562-35-5
http://dx.doi.org/10.14722/ndss.2014.23082

and in particular of the TMSI reallocation procedure. We highlight deficiencies in the standard and show how these have led to flawed implementations which do not trigger the reallocation procedure often enough, and when they do they sometimes allow linkability attacks. Our experimental analysis reveals some real and novel network scenarios which allow a third party to violate a user's privacy despite the reallocation protocol being used according to the current standard. In our formal analysis, we prove that the TMSI reallocation procedure provides unlinkability in case a new ciphering key is established before each execution of the TMSI reallocation procedure and we discuss other possible countermeasures. This proof is one of the few examples in the literature [4], [5] of a proof of labelled bisimilarity of a real-sized protocol. Our proof makes use of both manual and automatic proof techniques.

Terminology. In 3GPP specifications, mobile phones together with their SIM card are referred to as *mobile stations*, abbreviated MS. Mobile stations have a permanent identity stored in the SIM card, the *International Mobile Subscriber Identity*, abbreviated IMSI. As stated, the serving network (SN) assigns a temporary identity to an MS, called the *Temporary Mobile Subscriber Identity* (TMSI).

When the network wants to deliver a service to a mobile station (*e.g.* an incoming phone call) it sends a *paging request* message specifying the identity of the MS (TMSI or IMSI if the TMSI is not known). The paging request is sent on a common channel in all the locations most recently visited by the MS. A MS continuously monitors the common channel used for paging of the area it is located in. When the MS receives a paging request, it asks the base station it is attached to to assign a dedicated channel. The MS the sends a paging response containing its own identity (usually TMSI) in clear-text on the dedicated channel.

A. Related Work

Linkability of transactions has been identified and often reported by the media as an important threat to user privacy, in a variety of areas including on-line searches [6], road usage charging [7], electronic passports [8], and mobile telephony [3]. The problem of privacy is a multi-layer/multiprotocol problem [9] which requires all protocols at all layers to satisfy the desired properties. Moreover, privacy properties are often violated because of subtle design/implementation details, hence the need for careful analysis.

Most of the work on security of mobile telephony systems concerns content-secrecy, integrity and authentication properties [10], [11], [12]. There are only few formal and experimental studies concerning the level of usage-privacy provided to the user by mobile telephony systems. Foo Kune et al. [13] presented a study on the use of the paging procedure to locate mobile telephony users. They perform a tracking attack relying on passive sniffing of paging response messages triggered by placing silent phone calls (obtained by hanging up before the receiving phone rings) for the victim phone. This technique allows one to reveal the presence of the victim in an area monitored by the attacker. Munaut and Nohl [11] previously outlined a similar technique. They performed a GSM sniffing attack, which allows one to eavesdrop a GSM phone call by using a modification of the osmocom-BB [14] open source implementation of the GSM protocol stack and an old Motorola mobile phone. Differently from Foo Kune et al., they used a silent SMS to trigger the paging responses needed to locate the victim. Although these works take advantage of the fact that a TMSI is allocated for a long time window, they do not analyse the security and privacy provided by the TMSI reallocation procedure. Moreover, in order to perform the attack, the adversary needs to know the mobile number of the victim. Indeed, these attacks consist in establishing the presence of a target MS in a given location by linking the target's telephone number with its TMSI. This attack relies on the fact that TMSI reallocation is not activity-dependent (as confirmed by our experiments). This suggests the adoption of activity dependent reallocation strategies to thwart the attack. However, we show that reallocating a new TMSI after each transaction is not sufficient, because (as we experimentally show) encryption keys are reused in many deployed networks allowing the replay attack we present. This further privacy threat cannot be established from Foo Kune et al's analysis. We formally prove that establishing fresh keys at each TMSI reallocation and adopting an activity-dependent reallocation strategy thwarts Foo Kune et al's attack. Additionally, we show that deployed networks do not follow the standard as they do not all enforce TMSI reallocation at each change of Location Area. This makes a MS traceable across Location Areas by simple sniffing. This further privacy breach is beyond the scope of Foo Kune et al's analysis. So our findings further contribute to help improving future developments of this technology. The experiments we carried out show that real networks do not adopt policies for changing TMSI which are dependent on the number of exposure of the TMSI over-the-air by the mobile phone activity and hence they do not tackle these attacks.

Engel showed at the 25C3 conference [15] how network signalling messages, triggered when sending/receiving SMS messages, can be used to locate mobile telephony users. He suggests that network operators should use home routing, i.e. forwarding through the home network, as a countermeasure to this SMS tracking attack. This attack requires access to the intra-network communication infrastructure, which although possible may require subscription to a pay per query service. In this work, we analyse the privacy provided by the more exposed over-the-air communication available to any attacker with a radio enabled device and do not rely on the less easily accessible intra-network communication protocols.

The *gsmmap* project [16], [17] uses a variant of the open source GSM protocol stack developed within the osmocom-BB project to assess and visually render on a map the level of security and privacy provided by network operators across the world. In particular their aim is to check if network operators are protecting the users from well known attacks by adopting countermeasures such as the use of A5/3 encryption, padding randomization, and full authentication for outgoing calls and SMS to prevent impersonation and interception, and the use of regular TMSI updates, and home routing to prevent Engel's SMS tracking attack.

The closest work to ours is the one presented in [18] which also analyses mobile telephony protocols from a privacy

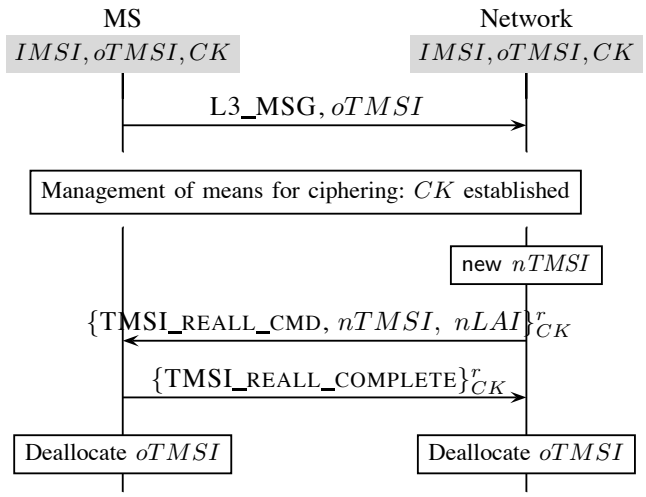

Fig. 1. TMSI Reallocation Procedure

point of view. Arapinis et al. [18] uncover some privacy attacks on the 3G authentication protocol and on the paging procedure. These attacks are exposed and exploited through a real implementation. The authors propose and automatically verify privacy-friendly fixes of the attacked procedures. The procedures analysed in [18] are not part of the identity management mechanisms of mobile telephony systems, in particular they do not analyse the TMSI reallocation procedure that is the procedure on which mobile telephony systems rely to provide anonymity and unlinkability from third parties. This procedure is the focus of our work. Moreover, in this work we are concerned with both issues of the standard specifications and issues of the actual implementation by real networks. None of the issues concerning the identity management and the pseudonym changing mechanism that we identify in this paper arise from the analysis presented in [18]. Finally, the proof methods used in [18] are too weak to prove the correctness of the TMSI reallocation. We have to create new proof techniques. In particular we combine both manual and automatic proofs in order to obtain the unlinkability proof sketched in Section IV-C.

II. Pseudonyms for User Privacy

A mobile station (MS) is uniquely identified by means of its IMSI. To avoid over-the-air attackers from identifying and linking a user's transactions, a temporary identity called TMSI is assigned by the network and is used to identify the mobile station in protocol messages. The mobile station identity (its TMSI, if available, or its IMSI) is always included in the first message sent from the MS to the network after the establishment of a dedicated channel. This allows the network to identify the MS before delivering a service to it. For example, the identity is carried in location update requests, CM (Call Management) requests, and paging responses. The use of TMSIs avoids the exposure of the long term unique identity (IMSI) and hence provides third-party anonymity to mobile telephony subscribers. The 3GPP standard specifies that a new TMSI should be assigned at least at each change of location area. Besides this constraint, the choice of how often a new assignment is performed within a location area is left to the network operators [19]. In order to prevent an adversary linking the old TMSI with the new one, the assignment of a new TMSI is performed in ciphered mode. The session key used to encrypt the new TMSI is established by executing the AKA protocol.

A. TMSI Reallocation Procedure

The TMSI reallocation procedure assigns a new pseudonym (TMSI) to a mobile station. The new TMSI is sent to the mobile station in an encrypted fashion. Figure 1 depicts the TMSI reallocation procedure as defined in the 3GPP standard [19], [20]:

- The mobile station sends a first message on a dedicated channel. This message contains the current MS's temporary identity $oTMSI$;
- on receipt of this message, the network can identify the MS and establish means for ciphering of the subsequent communication on the dedicated channel;
- the rest of the communication is then encrypted and consists of a TMSI reallocation command message containing a new pseudonym $nTMSI$ chosen by the network and the current location area $nLAI$ (the area within which $nTMSI$ is meaningful);
- this message is followed by a TMSI reallocation complete message which is sent by the MS to acknowledge the completion of the reallocation procedure.

If the network does not receive the expected acknowledgement from the MS, it maintains both $oTMSI$ and $nTMSI$ as valid pseudonyms for the the IMSI. The network can perform a TMSI reallocation at any time whilst a dedicated channel is established. The standard does not fully specify how often this procedure should be performed. However, it mandates that it should at least be performed at each change of location [19]. The standard defines two options for the management of the means for ciphering (i.e. to establish the ciphering key CK): (1) either a fresh ciphering key is established by executing the authentication procedure; (2) or a previously established ciphering key can be restored by means of the security mode set-up procedure, which allows the MS and the network to agree on a ciphering algorithm.

B. Subscriber Privacy Analysis

The 3GPP standard relies on frequent reallocation of TMSIs in order to provide user's untraceability. In particular, it mandates that TMSI reallocation should be performed whenever the MS moves between "location areas" (identified by location area identifiers, LAIs). However, it is known that location areas often extend over several square kilometres, and a subscriber's movements are typically confined within one or two location areas [21], [22]. So location areas may be too large to trigger TMSI reallocations in practice. Moreover, we show that one of the policies defined by the standard for the establishment of the ciphering key, namely the use of restored keys, allows a linkability attack on the TMSI reallocation procedure. In particular, we show that is adopted by real networks which makes their users vulnerable to tracking. Hence the

Fig. 2. Experimental Tools

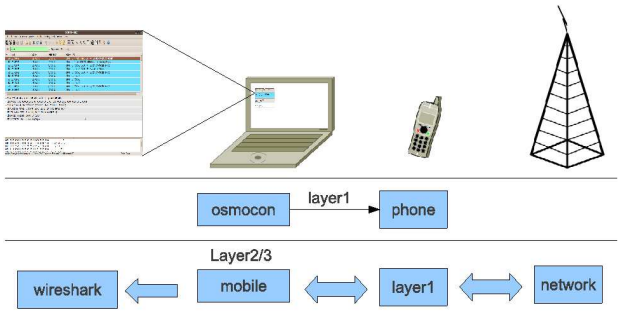

Fig. 3. Osmocom-BB architecture

standard should forbid the use of a previously established ciphering key for the execution of the TMSI reallocation procedure.

Section III reports on our experimental analysis. We monitored over-the-air communications of idle and active MSs in order to understand how real networks implement user identity confidentiality through the use of TMSIs, both in terms of frequency of reallocation, and ciphering keys used. Our experiments confirm that the reuse of previously established keys is a commonly adopted policy. However, we show that in case the reuse of encryption keys is adopted for the execution of the TMSI reallocation procedure, this enables a linkability attack which makes it possible to link old and new TMSIs.

In Section IV, we introduce the formal tools we use, and in Section IV-B the formal definition of unlinkability. In Section IV-C, we formally prove that using a fresh key for each TMSI reallocation would be enough to ensure users' privacy.

III. Experimental Analysis

Our experiments were carried out using an old GSM Motorola C115 mobile phone in France, UK, Greece, and Italy and using SIM cards from all the major UK, Greek, and Italian network operators.[1]

[1]More specifically, we used O2, T-Mobile, Vodafone, and Orange in the UK; Vodafone and Wind in Greece; Bouygues and Orange in France; and Wind, Vodafone and TIM in Italy.

A. Experimental Settings and Scenarios

The Motorola C115 has a TI Calypso baseband chipset which is supported by the Osmocom-BB project [14]. The Osmocom-BB project includes an open source implementation of the GSM baseband and various other applications aiming to implement a GSM mobile station. The radio communication functions are implemented in the firmware which is flashed from a laptop into the mobile phone through the Osmocon software, by means of a T191 unlock cable (Figure 2). The firmware implements layer 1 of the GSM protocol stack, while layers 2 and 3 are implemented in specialised applications running on the laptop and communicating with the mobile phone through the T191 cable (Figure 3). In particular, we used the 'mobile' application which implements layer 2 and 3 of the GSM protocol stack to provide all the basic functions of a mobile phone (network registration, location update, making and receiving calls, and sending and receiving SMSs).

The mobile phone activities are logged on a shell terminal and the radio communication is encapsulated in UDP packets sent to a configurable IP address. This traffic can be captured through the Wireshark network traffic analyser [23]. Interactions with the mobile phone are enabled by a telnet command interface. This allows one to manually select a network, start phone calls, send SMS and service requests, etc.

We captured over-the-air messages using the 'mobile' application in different settings: (1) mobile station in idle state and not moving; (2) mobile station in idle state and moving across two urban areas; (3) mobile station involved in activities such as receiving or starting phone calls, receiving or sending SMSs, and requesting services as for example call diversions.

Since the 3GPP standard merely gives guidelines, real networks differ in the implementation details of the TMSI reallocation. To understand if the different implementations achieve the privacy guarantees they were intended for, we analysed the traffic captured with the mobile application. In particular, we are interested in finding out if the frequency of TMSI reallocation execution is high enough to defeat passive and active tracking attacks, if the policy of changing TMSI at least at each change of location is actually implemented so to obtain at least location dependent privacy, and if the frequency of execution of the TMSI reallocation procedure is related to the amount of activity of the MS (i.e., to how often the TMSI is exposed to overhearing).

B. Findings/Results

We report on three different issues showing that some of the actual implementations of the strategy for changing pseudonyms to avoid tracking are not offering enough privacy guarantees to the mobile telephony subscribers. Our observation and their consequences on users' privacy are discussed in this section[2].

The TMSI reallocation procedure is rarely executed. Although in the standard the privacy offered to mobile phone bearers is based on frequent updates of TMSIs, our experiments show that the same TMSI can be allocated for

[2]The traces that allowed us to draw the conclusions presented are made available for inspection [24]

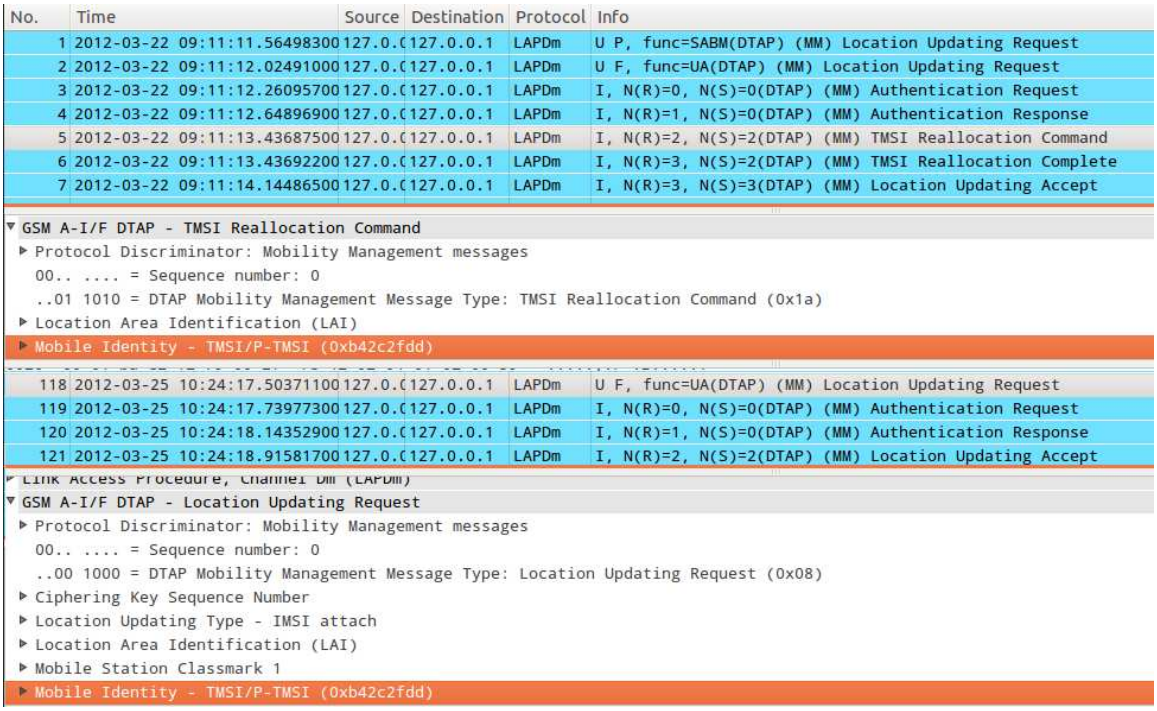

Fig. 4. Trace of a UK Vodafone SIM card obtaining a new TMSI (0xb42c2fdd) on 22/03/12. The same TMSI is still in use on 25/03/12 after 3 days from its allocation.

several hours and even days. Moreover, turning on and off the MS does not usually result in a new TMSI being allocated. As an example Figure 4 shows that a TMSI allocated on 22/03/2012 has not been updated by 25/03/2012, making the phone trackable for a period of 3 days. This behaviour can be observed for the major UK, Greek, French and Italian network operators. An attacker could take advantage of the long life of a TMSI and monitor a few sub-areas using short range devices in order to obtain a fine grained tracking of his victim within a same LAI.

We observed that the major UK network operators and the Vodafone and TIM Italian operators rarely execute the TMSI reallocation even in presence of MS activity, but the first message sent by a MS when requesting or receiving a service contains its TMSI, hence exposes it to eavesdropping third parties. As mentioned in Section I-A, TMSI liveness makes it possible to locate mobile telephony users without alerting them. This can be achieved by paging the victim and hence provoking a paging response. To reduce the set of answering TMSIs to the victim's one, the attacker must repeat the process several times because more than one MS could be sending a paging response at the same time and it is possible only if the TMSI is not reallocated even in case of activity exposing the TMSI (*e.g.*receiving calls). The attack in [13] thus relies on the low frequency of TMSI reallocations and demonstrates that changing pseudonyms, as mechanism to provide location privacy, is not effective without a policy for changing of pseudonyms which takes into account the actual exposure of the pseudonym caused by the mobile station activity.

A change of location area does not imply a change of TMSI although such a change is mandated by the 3GPP standard. We observed this behaviour when capturing the signalling messages of a mobile station moving by coach between different cities in the UK, using the Orange and the O2 networks where we observed the same pseudonym being accepted in different location areas with no further execution of the TMSI reallocation procedure. Assuming an average speed of 70Km/h we observed that a new TMSI was assigned after about 45 min (about 53km) and a second one after about 60 min (about 70km) while we observed a change of LAI every 5 min on average and hence a new TMSI should have been allocated, on average, about every 3km. Figure 5 shows an example trace where a TMSI used at location 234/33/1381 (packet no. 668) is accepted a different location 234/33/29 (packet no.678).

The fact that a TMSI was accepted in two neighbouring LAIs contradicts the specification that a TMSI reallocation should be performed at least at each change of location. However, changing pseudonym when changing location area would provide location-dependent privacy to the user since it would prevent passive tracking across different LAIs. The combination of the two behaviours reported so far (i.e. keeping the same TMSI for a long period of time and not changing it when changing location area) enables the attacker to both track his victim within an area and follow him across different areas without doing any extra effort other than passively sniffing.

Previously established keys are restored and used to encrypt the TMSI reallocation procedure. Our captures confirm that the reuse of previously established keys is a policy adopted by real networks and that in particular previously established keys are used for the execution of the TMSI real-location procedure. The experiments we performed show that

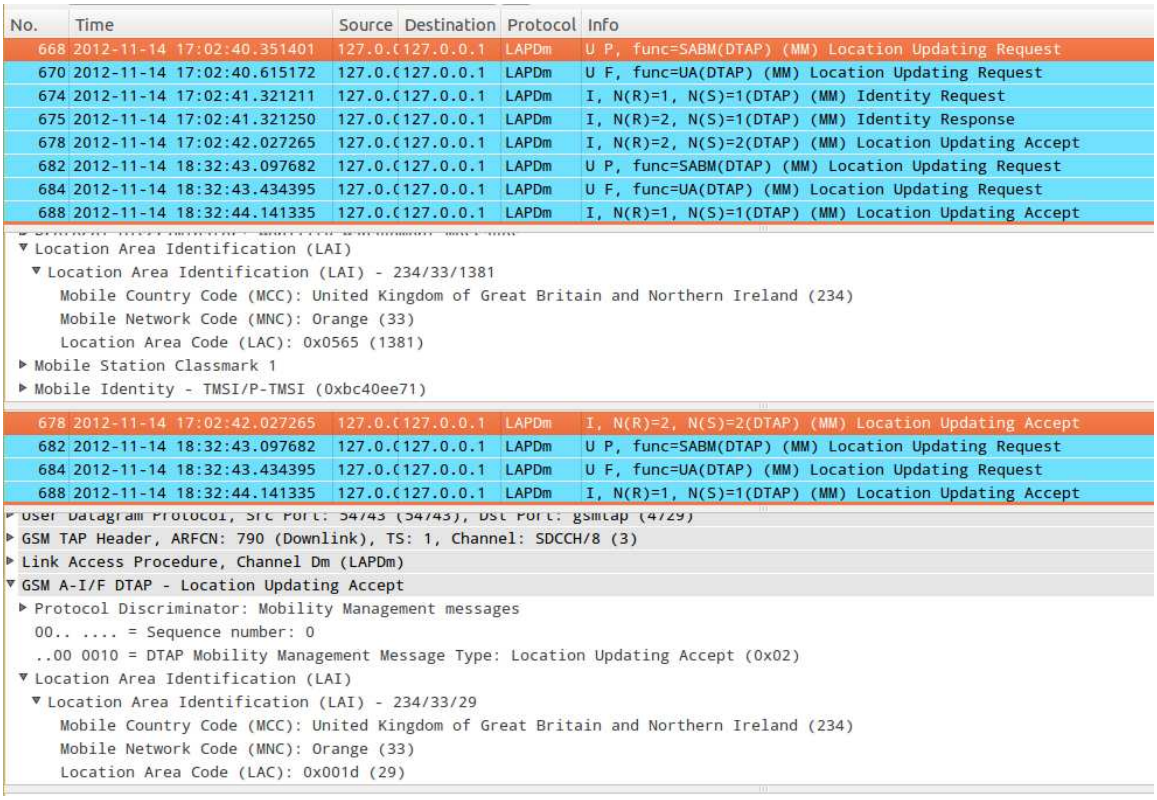

Fig. 5. Trace of a UK Orange SIM card. The TMSI used at location 234/33/1381 (packet no. 668) is accepted at location 234/33/29 (packet no.678), while the 3GPP standard mandates a TMSI reallocation at each change of location.

major UK and Italian network operators[3] reuse previously established keys instead of performing the authentication procedure before each execution of the TMSI reallocation procedure. Figure 7 shows a trace from a UK Lebara SIM card attached to the Vodafone network performing a location update (packet no. 4063). Then the execution of the authentication procedure establishes a new ciphering key (packets 4065, 4068) and consecutively the TMSI reallocation procedure (packets 4079, 4081) is executed. The subsequent TMSI reallocations (packets 9691, 9693, 71695, 71697, 92653, 92655) are executed without first performing the authentication procedure and hence reusing the previously established ciphering key.

The use of a previously established ciphering key enables replay attacks such as the one depicted in Figure 6. An attacker, controlling a radio device able to sniff and inject messages over-the-air, first captures a TMSI reallocation command (the second message in Figure 6). Later on, when the MS has possibly already changed its pseudonym but not yet established a new encryption key, the attacker can replay the captured TMSI reallocation command (one message before last in Figure 6). The victim's MS successfully decrypts the reallocation message and sends the TMSI reallocation complete message. This allows the attacker to distinguish the victim's MS from any other that would not successfully decrypt the message and thus would not send any reply, even though in the meantime a different TMSI ($nTMSI_k$ in Figure 6) was assigned to the victim's MS. For example, the TMSI reallocation packet no. 71695 in Figure 7 does not achieve its goal since, by executing the above-mentioned attack, an attacker could link the newly assigned TMSI with the previously allocated one (packet no. 4079).

This attack would not be possible if a new ciphering key CK' was established. In this case, the replayed reallocation message sent from the adversary and previously encrypted with the key CK could not be decrypted by the victim mobile phone using key CK' and hence the reallocation would fail. The adversary cannot deduce any information from this since it does not know if the procedure failed because the key was changed or because the mobile phone is not the victim one.

Two realistic adversary scenarios for the TMSI reallocation attack could be the profiling of user in a defined urban area or the tracking of a target victim in few selected areas.

An attacker interested in profiling user's movements in a specific area (say to few square kilometres) can use our attack to trace users' movements in the area across different days. This attacker could use a set of short to medium range devices (from 10m to 1km), requiring an investment of a few thousand dollars.

An attacker with a more limited budget interested in tracking a specific target in few sensitive locations (imagine a stalker or jealous partner or over-controlling employer) probably knows his/her victim and his/her habits. For this purpose short range devices could be used (with an investment of a few hundred dollars).

[3]UK: Vodafone and T-mobile; Italy: Vodafone.

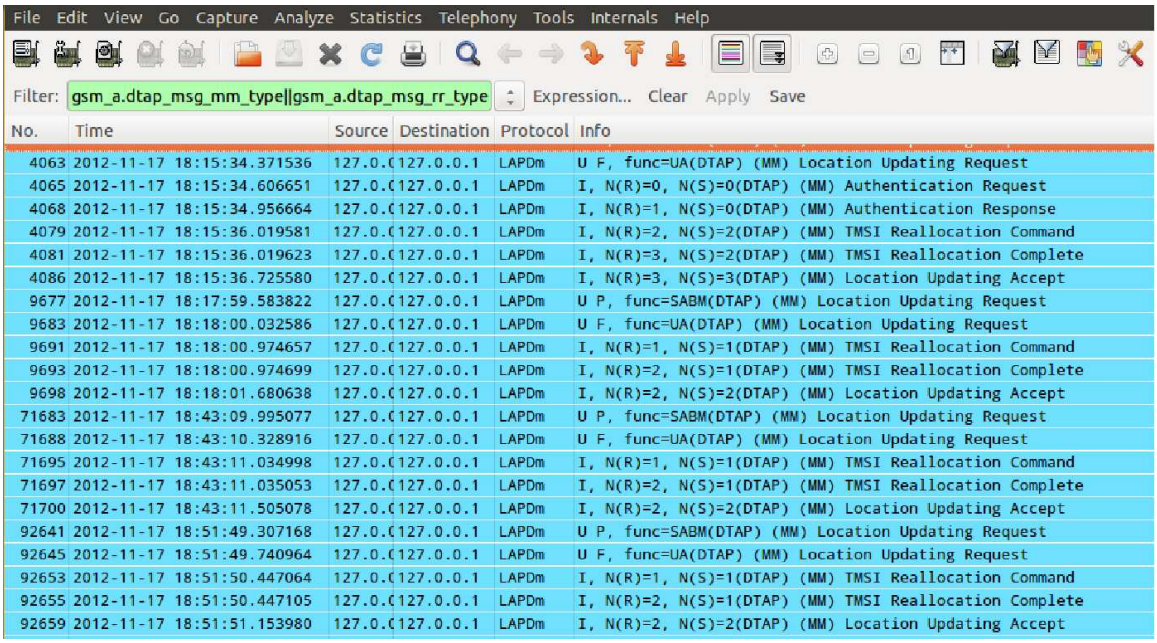

Fig. 7. Trace of a UK Lebara SIM card attached to the Vodafone network while travelling on a train. The TMSI reallocation procedure is executed by reusing a previously established key. The MS first performs a location update (packet no. 4063), then the authentication procedure to establish a ciphering key (packets 4065, 4068), followed by the TMSI reallocation procedure (packets 4079, 4081). The following three TMSI reallocations (packets 9691, 9693, 71695, 71697, 92653, 92655) are executed without first performing the authentication procedure and hence reusing the previously established ciphering key.

IV. Formal Analysis

Often, deployed protocols are subsequently found to be flawed and to be subject of attacks. In this paper we showed that the possibility of restoring the ciphering key CK enables a linkability attack on the TMSI reallocation procedure. This weakness was hidden in the protocol logic and was not evident to the protocol designers. This demonstrates that rigorous formal analysis is needed to (1) give strong guarantees on the properties achieved by security protocols, and (2) clearly assess the assumptions under which these properties hold. We formally analyse the TMSI reallocation procedure w.r.t. a rigorous definition of unlinkability as given by Arapinis et al. in [25]. In particular, we model the TMSI reallocation procedure using the applied pi-calculus. We prove that if new session keys are established before each execution of the TMSI reallocation procedure, then the attack presented in the previous section is thwarted, and the subscriber's unlinkability is preserved by the protocol.

A. Applied pi-Calculus

The applied pi-calculus is a formal language, introduced by Abadi and Fournet [26], for modelling concurrent processes and in particular to ease the reasoning about cryptographic protocols. In the applied pi-calculus, cryptographic primitives are modelled as functions and messages are represented by *terms* L, M, N, T built over an infinite set of names $a, b, c, \ldots$ an infinite set of variables $x, y, z, \ldots$ and a finite set of function symbols $f(M_1, \ldots, M_l) \in \Sigma$ (which includes the considered cryptographic primitives). A function symbol with arity 0 is a constant symbol. Function properties are modelled by means of a set of equations defining an equational theory E on the set of possible terms. We define equality modulo the equational theory, written $=_E$, as the smallest equivalence relation on terms, that contains E and is closed under application of contexts, substitution of terms for variables and bijective renaming of names.

Syntax. The grammar for *processes* of the applied pi-calculus is the following:

$P, Q, R ::=$	plain processes
0	null
$P \mid Q$	parallel
$!P$	replication
$\nu\, n.P$	restriction
if $M = N$ *then* P *else* Q	conditional
$c(x).P$	input
$\bar{c}\langle M\rangle.P$	output
$A, B, C ::=$	extended processes
P	plain process
$A \mid B$	parallel
$\nu\, n.A$	name restriction
$\nu x.A$	variable restriction
$\{^M/_x\}$	active substitution

The null process does nothing. The parallel composition of P and Q represents the parallel execution of P and Q. The replication of a process P acts like the parallel execution of an unbounded number of copies of P. The name restriction $\nu n.P$ creates a new name n whose scope is restricted to the process P and then runs P. The `if` construct defines a process that evaluates the condition $M = N$ and behaves as P, if $M =_E N$, and otherwise behaves as Q. Note that we check for equality modulo the equational theory rather than syntactic equality of terms. The message input $c(x).P$ represents a process ready to input from the channel c, the actual message received will be substituted to x in P. The syntactic

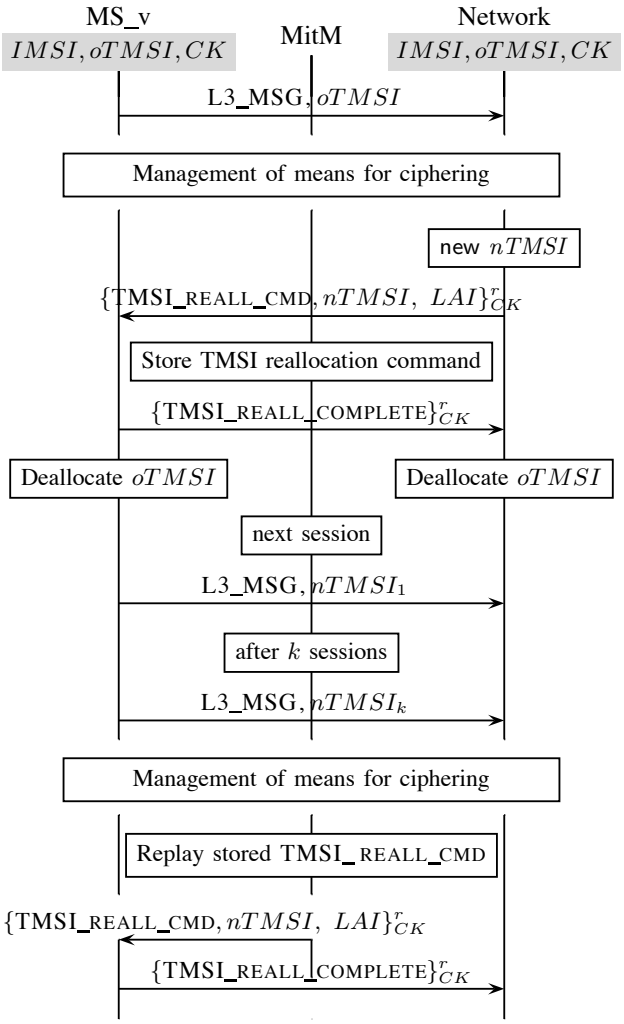

Fig. 6. TMSI Reallocation Procedure Attack

substitution of a term T for the variable x in the process P is denoted by $P\{^T/_x\}$. The message output $\bar{c}\langle M\rangle.P$ describes a process ready to send a term M on the channel c and then to run P. Extended processes are introduced to represent the adversarial knowledge. They include plain processes, parallel compositions, bindings of names and variables and active substitutions. Active substitutions represent the knowledge acquired by the adversary as result of the process execution, in particular the active substitution $\{^M/_x\}$ represent the fact that the adversary can access the term M through the handle x. We define the set of bounded (resp. free) names $bn(A)$ (resp. $fn(A)$) of a process A as the set containing every name n which is under restriction $\nu\, n$ (resp. not under the scope binder $\nu\, n$) in A . The set of bound (resp. free) variables $bv(A)$ (resp. $fv(A)$) of A consists of all those variables x bound by restriction $\nu\, x$ or input $u(x)$ (resp. not under the scope of a restriction $\nu\, x$ or input $u(x)$) in A. Similarly we define $fn(M)$ and $fv(M)$, for the set of free names, respectively variables, which appear in the term M. We say that an extended process is closed when every variable x is either bound or defined by an active substitution $\{^M/_x\}$ for some term M. A frame, ϕ, is an extended process built from 0 and active substitutions $\{^M/_x\}$ composed by parallel composition and restriction. The domain $dom(\phi)$ of a frame is the set of variables x for which ϕ contains an active substitution $\{^M/_x\}$ such that x is not under restriction. The frame $\phi(A)$ of a process A is obtained by replacing every plain process in A with 0. The frame $\phi(A)$ represents the knowledge A outputs to its environment. The domain $dom(A)$ of A is the domain of $\phi(A)$.

Example 1: Using functions and equations we can define randomized symmetric encryption and pairing. Let $\Sigma = \{\mathtt{senc}/3,\ \mathtt{sdec}/2, \mathtt{pair}/2, \mathtt{fst}/1, \mathtt{snd}/1\}$, and consider the equations:
$\mathtt{sdec}(k, \mathtt{senc}(k, r, m)) = m$, $\mathtt{fst}(\mathtt{pair}(x, y)) = x$, $\mathtt{snd}(\mathtt{pair}(x, y)) = y$. The first equation allows to decrypt an encrypted message, m, given the knowledge of the encryption key k. This is the usual rule to model randomized symmetric encryption. The rest of the rules allow to decompose a pair and retrieve its components.

As example of processes, we introduce MS and SN modelling respectively a mobile station and a serving network sharing a private channel dck. This private channel models the fact that MS and SN can "securely" establish a shared session key by executing the authentication procedure. The private channel d models a mobile station's memory (or state) recording the currently assigned TMSI. Input messages are read from the dw channel and output messages are sent on the up channel. We consider an attacker that intercepts all communications on public channels.

$$
\begin{aligned}
Init &\stackrel{def}{=} \overline{d}\,\langle id\rangle\\
MS &\stackrel{def}{=} \nu\, ck.\nu\, mr.d(x).\overline{up}\langle x\rangle.\overline{dck}\langle ck\rangle.dw(y).\\
&\quad \textit{if}\ \ \mathtt{fst}(\mathtt{sdec}(ck, y)) = \text{TMSI\_REALL}\ \ \textit{then}\\
&\quad\quad \overline{up}\langle \mathtt{senc}(ck, mr, \text{COMPLETE})\rangle.\\
&\quad\quad \overline{d}\langle \mathtt{snd}(\mathtt{sdec}(ck, y))\rangle\\
&\quad \textit{else}\ 0\\
SN &\stackrel{def}{=} \nu\, nid.\nu\, sr.dw(z).dck(xck).\\
&\quad \overline{up}\langle \mathtt{senc}(xck, sr, \mathtt{pair}(\text{TMSI\_REALL},\ nid))\rangle.\\
&\quad dw(w)\\
M &\stackrel{def}{=} \nu\, dck.(!(\nu\, d.\nu\, id.(Init\ |!MS))\ |!SN)
\end{aligned}
$$

The $Init$ process initializes the MS memory by storing in it the initial pseudonym id. The current pseudonym is stored in the memory d and is sent by the MS with its first message. The MS then establishes a session key with the network, modelled here by the communication of a new key ck over a private channel dck (note that in this model a fresh session key is established before the execution of each TMSI reallocation). The mobile station then receives a message and checks if it is a legitimate TMSI reallocation command message encrypted by the network using the session key (*if* $\mathtt{fst}(\mathtt{sdec}(ck, y)) =$ TMSI_REALL). In this case it sends a TMSI reallocation complete message ($\overline{up}\langle \mathtt{senc}(ck, mr,\ \text{COMPLETE})\rangle$) and updates its own memory with the new pseudonym received in the TMSI Reallocation command ($\overline{d}\langle \mathtt{snd}(\mathtt{sdec}(ck, y))\rangle$). For simplicity, we do not model the eventual updating of the location area.

Structural Equivalence. The structural equivalence relation defines when syntactically different processes actually represent the same process, for example by equating $A \mid B$ and $B \mid A$ which both represent the parallel execution of A and B.

Formally, structural equivalence ($\equiv$) is the smallest equivalence relation on extended processes that is closed by α-conversion of both bound names and bound variables, and closed under application of evaluation contexts such that:

$A \equiv A \mid 0$	PAR-0
$A \mid (B \mid C) \equiv (A \mid B) \mid C$	PAR-A
$A \mid B \equiv B \mid A$	PAR-C
$!P \equiv P \mid !P$	REPL
$\nu\, n.0 \equiv 0$	NEW-0
$\nu\, u.\nu\, w.A \equiv \nu\, w.\nu\, u.A$	NEW-C
$A \mid \nu\, u.B \equiv \nu u.(A \mid B)$ where $u \in fv(A) \cup fn(A)$	NEW-PAR
$\nu x.\{^M/_x\} \equiv 0$	ALIAS
$\{^M/_x\} \mid A \equiv \{^M/_x\} \mid A\{^M/_x\}$	SUBST
$\{^M/_x\} \equiv \{^N/_x\}$ where $M =_E N$	REWRITE

The rules for parallel composition, replication and restriction are easy to understand and capture our intuition of the operators properties. The ALIAS rule allows to introduce arbitrary active substitutions with restricted scope. SUBST allows the application of an active substitution to a process running in parallel with it. REWRITE, allows the substitution of terms that are equal modulo the equational theory.

Semantics. The *internal reduction relation* ($\xrightarrow{\tau}$) describes how processes evolve in isolation. Formally, internal reduction is the smallest relation on extended processes closed by structural equivalence and application of evaluation contexts such that:

$\overline{c}\langle M\rangle.P \mid c(x).Q$	$\xrightarrow{\tau}$	$P \mid Q\{^M/_x\}$		Comm
if $M = N$ *then* P *else* Q	$\xrightarrow{\tau}$	P	if $M =_E N$	Then
if $M = N$ *then* P *else* Q	$\xrightarrow{\tau}$	Q	if $M \neq_E N$	Else

Input and output actions on a channel c can be synchronized, resulting in the communication of the term M through the handle x. The `if` construct evaluates the equality modulo the equational theory between two terms M and N and executes the process P or the process Q accordingly. The evaluation of M and N may require the application of all the active substitutions in order to obtain the ground equivalent (i.e. containing no variables) of the terms M and N. We denote with $\Rightarrow$ the reflexive and transitive closure of $\xrightarrow{\tau}$.

The *labelled reduction relation* ($\xrightarrow{\alpha}$) describes how processes interact with the environment. The label α is either an input, an output, or a restricted output. The labelled reduction relation extends the internal reduction enabling interactions with the environment as defined by the following rules:

$$c(x).P \xrightarrow{c(M)} P \qquad \text{(INPUT)}$$

$$\overline{c}\langle u\rangle.P \xrightarrow{\overline{c}\langle u\rangle} P \qquad \text{(OUT-ATOM)}$$

$$\frac{A \xrightarrow{\overline{c}\langle u\rangle} A',\ u \neq c}{\nu u.A \xrightarrow{\nu u.\overline{c}\langle u\rangle} A'} \qquad \text{(OPEN-ATOM)}$$

$$\frac{A \xrightarrow{\alpha} A',\ u \text{ does not occur in } \alpha}{\nu u.A \xrightarrow{\alpha} \nu u.A'} \qquad \text{(SCOPE)}$$

$$\frac{A \xrightarrow{\alpha} A',\ bv(\alpha) \cap fv(B) = bn(\alpha) \cap fn(B) = \emptyset}{A \mid B \xrightarrow{\alpha} A' \mid B} \qquad \text{(PAR)}$$

$$\frac{A \equiv B,\ B \xrightarrow{\alpha} B',\ B' \equiv A'}{A \xrightarrow{\alpha} A'} \qquad \text{(STRUCT)}$$

The INPUT rule allows a process to input a term from the environment through its handle x. OUT-ATOM allows a process to output a variable or a channel name, while OPEN-ATOM enables the output of a restricted variable. The SCOPE rule says that the scope of names and variables not involved in the labelled transition is preserved by the transition relation. The rule PAR allows one of the processes involved in a parallel composition to evolve. The rule STRUCT states that the labelled transition relation is closed under structural equivalence.

Equivalence Relations. Static equivalence defines classes of processes having released equivalent knowledge to the environment. It only looks at the current state of the processes, not at their possible evolutions.

Definition 1 (Static Equivalence): Two closed frames $\phi \equiv \nu\, \tilde{n}.\sigma$ and $\psi \equiv \nu\, \tilde{n}.\tau$ are statically equivalent, denoted $\phi \approx_s \psi$, if $dom(\phi) = dom(\psi)$ and for all terms M, N such that $\tilde{n} \cap (fn(M) \cup fn(N)) = \emptyset$, we have that $M\sigma =_E N\sigma$ holds if and only if $M\tau =_E N\tau$ holds. Two closed extended processes A, B are statically equivalent, $A \approx_s B$, if $\phi(A) \approx_s \phi(B)$.

The labelled bisimilarity relation defines classes of processes whose interactions with the environment are equivalent at each step for any possible evolution of the processes. Intuitively, two processes are labelled bisimilar if one can mimic the actions of the other step by step outputting the equivalent knowledge to the environment at each step and vice versa.

Definition 2 (Labelled Bisimilarity): Labelled bisimilarity ($\approx_l$) is the largest symmetric relation $\mathcal{R}$ on closed extended processes such that $A\mathcal{R}B$ implies:

- $A \approx_S B$
- if $A \xrightarrow{\tau} A'$ then $\exists\, B'$ such that $B \Rightarrow B'$ and $A'\mathcal{R}B'$
- if $A \xrightarrow{\alpha} A'$ and $fv(\alpha) \subseteq dom(A)$ and $bn(\alpha) \cap fn(B) = \emptyset$; then $\exists\, B'$ such that $B \Rightarrow \xrightarrow{\alpha} \Rightarrow B'$ and $A'\mathcal{R}B'$.

B. Unlinkability of the Fixed TMSI Reallocation Procedure

Thanks to the equivalence relations defined above we can define various security properties. In particular, we can use labelled bisimilarity to state a property in terms of undistinguishability of a process from some ideal version of it, *i.e.* a version which satisfies the required property by construction. This is the idea behind the definition of unlinkability proposed by Arapinis, Chothia, Ritter and Ryan in [25].

Intuitively, such definition requires an ideal system, P_{UNLINK}, where each agent can execute the protocol at most once (and hence is unlinkable by construction) to be undistinguishable from a system, P, where each agent can execute the protocol an unbounded number of times. Formally:

Definition 3 (Strong Unlinkability): Let Σ be a signature and E an equational theory for this signature, and let P be a protocol over Σ of the form $P = !(\nu\ \tilde{m}.init.\ !main\_protocol)$. We build the protocol $P_{UNLINK} = !(\nu\ \tilde{m}.init.\ main\_protocol)$. We say that P preserves strong unlinkability if $P \approx_l P_{UNLINK}$

The definition of strong unlinkability allows us to formally analyse the TMSI reallocation procedure and establish if it achieves the desired unlinkability property when a new session key is established prior to each execution of the TMSI reallocation procedure.

Let the $Init, MS$ and SN processes be as defined in Example 1. We define:

$$\begin{aligned} SSA &\stackrel{def}{=} \nu\, d.\nu\, id.(Init \mid MS) \\ MSA &\stackrel{def}{=} \nu\, d.\nu\, id.(Init \mid !MS) \end{aligned}$$

The processes SSA and MSA are respectively a single-session and a multi-session mobile station agent. Single-session mobile stations can only execute one session of the TMSI reallocation procedure hence are unlinkable by construction and are part of the ideal system, while the multi-session agents represent the mobile stations of the real systems i.e. the ones we want to prove to be unlinkable, although they can execute several sessions of the procedure.
Let S and M be the two closed processes defined as follows:

$$\begin{aligned} S &\stackrel{def}{=} \nu\, dck.(!SSA \mid !SN) \\ M &\stackrel{def}{=} \nu\, dck.(!MSA \mid !SN) \end{aligned}$$

The process S represents an unbounded number of mobile stations executing the TMSI reallocation procedure at most once. The process M represents an unbounded number of mobile stations which can execute the TMSI reallocation procedure an unbounded number of times. We want to prove that M and S are labelled bisimilar and hence that M satisfies unlinkability.

However, the presence of the memory (state) for the storage of the currently assigned identity makes the automatic verification of the TMSI reallocation procedure not feasible with the `ProVerif` tool [27], which is to date the only tool able to automatically verify observational equivalence based properties for unbounded processes like the ones considered in this work. In fact, ProVerif cannot prove the observational equivalence of the following toy processes which model one process sending a fresh name on a public channel and another reading a fresh name from its state (modelled by the private channel d) and then sending it on a public channel:

$$\begin{aligned} &\nu\, d.!(\nu\ n.d(x).\tilde{d}\langle n\rangle.\tilde{c}\langle n\rangle \mid \nu\ m.\tilde{d}\langle m\rangle) \\ &\nu\, d.!(\nu\ n.d(x).\tilde{d}\langle n\rangle.\tilde{c}\langle x\rangle \mid \nu\ m.\tilde{d}\langle m\rangle) \end{aligned}$$

This happens because the abstractions ProVerif does for the sake of termination allow the process using the private channel to never consume the input. Hence, once a name is sent on the private channel `d`, that name can be read from it again and again, making the two processes not observationally equivalent. This is one of the reasons that led to the development of StatVerif [28], an extension of ProVerif which deals with stateful processes. However, StatVerif is not suitable in our case since it does not yet handle observational equivalence. For this reason we carry out a manual analysis instead. In the next section we give a sketch of the proof of the unlinkability of the TMSI reallocation procedure (Proposition 1) when performed by establishing a fresh session key prior to each execution.

C. Unlinkability Proof Sketch

To be able to describe the relation $\mathcal{R}$ witnessing that $S \approx_l M$ we define partial execution steps of the multi (resp. single)-session process's components as specified below. The process $MMS^k_{i,j}$ represents the i^{th} mobile station executing the k^{th} step of its j^{th} session of the TMSI reallocation protocol, while the process $SMS^k_{i,j}$ represents the $(i+j)^{th}$ mobile station executing the k^{th} step of its unique session. The process SN^l_m represents the l^{th} step of the m^{th} session of the serving network. The key point of the proof is to show that processes $MMS^k_{i,j}$ and $SMS^k_{i,j}$ simulate each other. We now give an outline of how this simulation works, by explaining how to match transitions in the multi-session and single-session processes.

1) Any transition within a session of some mobile station is a transition from $MMS^k_{i,j}$ to $MMS^{k'}_{i,j}$, with $k' > k$. There is always a matching transition within the single session of the corresponding mobile station from $SMS^k_{i,j}$ to $SMS^{k'}_{i,j}$, and vice versa.
2) The transitions for the serving network are the same in the multi-session an the single-session process, hence they match trivially.
3) The start of a new session for the same mobile station is modelled by a transition from $MMS^6_{i,j} \mid MS$ to $MMS^7_{i,j} \mid MMS^0_{i,j+1}$. The corresponding transitions in the single-session process, which are $SMS^6_{i,j}$ to $SMS^7_{i,j}$ and $Init \mid MS$ to $SMS^0_{i,j+1}$, model the use of an additional mobile station to simulate this extra-session.
4) The use of an additional mobile station in the multiple session process is modelled by a transition from $Init|MS$ to $MMS^0_{i+1,1}$. There is always a matching transition from $Init|MS$ to $SMS^0_{i+1,1}$ in the single-session process, and vice versa.

So far, this produces a perfect match between transitions for the multiple-session process and the single-session processes in cases 1, 2 and 4. In case 3, we still have to find a matching transition for the transition from $Init \mid MS$ to $SMS^0_{i,j+1}$

without $SMS^6_{i,j}$ being present. In this case we use the fact that $SMS^0_{i,j+1}$ and $SMS^0_{i+1,1}$ are α-equivalent and use case 4 to find a matching transition in the multi-session process. This point is the key part of the proof and shows that the single-session system really models several sessions of the same mobile station by using several mobile stations.

We now present this proof in more detail. We start by defining matching pairs of multi and single-session mobile stations for each evolution step k.

Let $i,\ j \in \mathbb{N}$. We denote:

$$Init_{i,j} \stackrel{def}{=} \overline{d_{i,j}}\langle id_{i,j}\rangle$$

$$MChk_{i,j} \stackrel{def}{=} \textit{if } \mathtt{fst}(\mathtt{sdec}(ck_{i,j},\ y_{i,j})) = \text{TMSI\_REALL } \textit{then}\ \overline{up}\langle \mathtt{senc}(ck_{i,j},\ mr_{i,j},\ \text{COMPLETE})\rangle.\ \overline{d_{i,1}}\langle \mathtt{snd}(\mathtt{sdec}(ck_{i,j},\ y_{i,j}))\rangle\ \textit{else } 0$$

$$SChk_{i,j} \stackrel{def}{=} \textit{if } \mathtt{fst}(\mathtt{sdec}(ck_{i,j}, y_{i,j})) = \text{TMSI\_REALL } \textit{then}\ \overline{up}\langle \mathtt{senc}(ck_{i,j},\ mr_{i,j},\ \text{COMPLETE})\rangle.\ \overline{d_{i,j}}\langle \mathtt{snd}(\mathtt{sdec}(ck_{i,j},\ y_{i,j}))\rangle\ \textit{else } 0$$

$$MMS^0_{i,j} \stackrel{def}{=} d_{i,1}(x_{i,j}).\overline{up}\langle x_{i,j}\rangle.\overline{dck}\langle ck_{i,j}\rangle.dw(y_{i,j}).MChk_{i,j}$$
$$SMS^0_{i,j} \stackrel{def}{=} d_{i,j}(x_{i,j}).\overline{up}\langle x_{i,j}\rangle.\overline{dck}\langle ck_{i,j}\rangle.dw(y_{i,j}).SChk_{i,j}$$
$$MMS^1_{i,j} \stackrel{def}{=} \overline{up}\langle M_{i,j}\rangle.\overline{dck}\langle ck_{i,j}\rangle.dw(y_{i,j}).MChk_{i,j}$$
$$SMS^1_{i,j} \stackrel{def}{=} \overline{up}\langle id_{i,j}\rangle.\overline{dck}\langle ck_{i,j}\rangle.dw(y_{i,j}).SChk_{i,j}$$
$$MMS^2_{i,j} \stackrel{def}{=} MX_{i,j} \mid \overline{dck}\langle ck_{i,j}\rangle.dw(y_{i,j}).MChk_{i,j}$$
$$SMS^2_{i,j} \stackrel{def}{=} SX_{i,j} \mid \overline{dck}\langle ck_{i,j}\rangle.dw(y_{i,j}).SChk_{i,j}$$
$$MMS^3_{i,j} \stackrel{def}{=} MX_{i,j} \mid dw(y_{i,j}).MChk_{i,j}$$
$$SMS^3_{i,j} \stackrel{def}{=} SX_{i,j} \mid dw(y_{i,j}).SChk_{i,j}$$
$$MMS^4_{i,j} \stackrel{def}{=} MX_{i,j} \mid MChk_{i,j}\{^{N_{i,j}}/_{y_{i,j}}\}$$
$$SMS^4_{i,j} \stackrel{def}{=} SX_{i,j} \mid SChk_{i,j}\{^{N_{i,j}}/_{y_{i,j}}\}$$
$$MMS^5_{i,j} \stackrel{def}{=} MX_{i,j} \mid \overline{up}\langle \mathtt{senc}(ck_{i,j},\ mr_{i,j},\ \text{COMPLETE})\rangle.\ \overline{d_{i,1}}\langle \mathtt{snd}(\mathtt{sdec}(ck_{i,j},\ N_{i,j}))\rangle$$
$$SMS^5_{i,j} \stackrel{def}{=} SX_{i,j} \mid \overline{up}\langle \mathtt{senc}(ck_{i,j},\ mr_{i,j},\ \text{COMPLETE})\rangle.\ \overline{d_{i,j}}\langle \mathtt{snd}(\mathtt{sdec}(ck_{i,j},\ N_{i,j}))\rangle$$
$$MMS^6_{i,j} \stackrel{def}{=} MX_{i,j} \mid MK_{i,j} \mid \overline{d_{i,1}}\langle \mathtt{snd}(\mathtt{sdec}(ck_{i,j},\ N_{i,j}))\rangle$$
$$SMS^6_{i,j} \stackrel{def}{=} SX_{i,j} \mid SK_{i,j} \mid \overline{d_{i,j}}\langle \mathtt{snd}(\mathtt{sdec}(ck_{i,j},\ N_{i,j}))\rangle$$
$$MMS^7_{i,j} \stackrel{def}{=} MX_{i,j} \mid MK_{i,j} \mid 0$$
$$SMS^7_{i,j} \stackrel{def}{=} SX_{i,j} \mid SK_{i,j} \mid \overline{d_{i,j}}\langle \mathtt{snd}(\mathtt{sdec}(ck_{i,j},\ N_{i,j}))\rangle$$

$$MMS^8_{i,j} \stackrel{def}{=} MX_{i,j} \mid 0$$
$$SMS^8_{i,j} \stackrel{def}{=} SX_{i,j} \mid 0$$

$$MX_{i,j} \stackrel{def}{=} \{^{M_{i,j}}/_{x_{i,j}}\}$$
$$SX_{i,j} \stackrel{def}{=} \{^{id_{i,j}}/_{x_{i,j}}\}$$

$$SK_{i,j}, MK_{i,j} \stackrel{def}{=} \{^{\mathtt{senc}(ck_{i,j},\ mr_{i,j},\ \text{COMPLETE})}/_{k_{i,j}}\}$$

$$RMS_i \stackrel{def}{=} \nu\, ck.\nu\, mr.(d_{i,1}(x).\overline{up}\langle x\rangle.\overline{dck}\langle ck\rangle.dw(y).\ \textit{if } \mathtt{fst}(\mathtt{sdec}(ck,\ y)) = \text{TMSI\_REALL } \textit{then}\ \overline{up}\langle \mathtt{senc}(ck,\ mr,\ \text{COMPLETE})\rangle.\ \overline{d_{i,1}}\langle \mathtt{snd}(\mathtt{sdec}(ck,\ y))\rangle\ \textit{else } 0$$

$$M_{i,j} \stackrel{def}{=} \begin{cases} id_{i,j} & \text{if } j = 1 \\ nid_{i,j-1} & \text{otherwise} \end{cases}$$

$$\widetilde{ss}_{i,j} \stackrel{def}{=} id_{i,1},\ d_{i,1},\ ck_{i,1},\ mr_{i,1}, \ldots, id_{i,j},\ d_{i,j},\ ck_{i,j},\ mr_{i,j}$$
$$\widetilde{ms}_{i,j} \stackrel{def}{=} id_{i,1},\ d_{i,1},\ ck_{i,1},\ mr_{i,1}, \ldots,\ ck_{i,j},\ mr_{i,j}$$

Note that a full execution of the TMSI reallocation procedure by a multi (resp. single)-session mobile station goes through the first 6 evolution steps. In particular, a new session of the TMSI reallocation protocol can be executed (by the multi-session MS) only after the mobile station fully completed the previous session ending up at step $k = 6$ where the synchronization on the memory channel d is enabled by the output of the newly allocated temporary identity $\overline{d_{i,1}}\langle \mathtt{snd}(\mathtt{sdec}(ck_{i,j},\ N_{i,j}))\rangle$. In case the *if* condition is not satisfied both the multi and the single-session mobile stations end up in a deadlock state ($k = 8$).

$$SN^0_i \stackrel{def}{=} \nu\, nid_i.\nu\, sr_i.dw(z_i).dck(xck_i).\ \overline{up}\langle \mathtt{senc}(xck_i,\ sr_i,\ \mathtt{pair}(\text{TMSI\_REALL},\ nid_i))\rangle.\ dw(w_i)$$
$$SN^1_i \stackrel{def}{=} \nu\, nid_i.\nu\, sr_i.dck(xck_i).\ \overline{up}\langle \mathtt{senc}(xck_i,\ sr_i,\ \mathtt{pair}(\text{TMSI\_REALL},\ nid_i))\rangle.\ dw(w_i)$$
$$SN^2_i \stackrel{def}{=} \overline{up}\langle \mathtt{senc}(xck_i,\ sr_i,\ \mathtt{pair}(\text{TMSI\_REALL},\ nid_i))\rangle.\ dw(w_i)$$

$$SN^3_i \stackrel{def}{=} \{^{\mathtt{senc}(ck_i,\ sr_i,\ \mathtt{pair}(\text{TMSI\_REALL},\ nid_i))}/_{y_i}\} \mid dw(w_i)$$
$$SN^4_i \stackrel{def}{=} \{^{\mathtt{senc}(ck_i,\ sr_i,\ \mathtt{pair}(\text{TMSI\_REALL},\ nid_i))}/_{y_i}\}$$

$$SN^k_{i,j} \stackrel{def}{=} SN^k_l\{^{ck_{i,j}}/_{xck_l},\ ^{y_{i,j}}/_{y_l},\ ^{w_{i,j}}/_{w_l},\ ^{nid_{i,j}}/_{nid_l},\ ^{sr_{i,j}}/_{sr_l}\},\ k \geq 2$$

$$\widetilde{nid}_{i,j} \stackrel{def}{=} nid_{i,1},\ sr_{i,1}, \ldots nid_{i,j}\ sr_{i,j}$$

$MX_{i,j}, MK_{i,j}$, and SN^4_i (resp. $SX_{i,j}, SK_{i,j}$ and SN^4_i) are the possible active substitutions resulting from one full execution of the TMSI reallocation procedure by the j^{th} session of the i^{th} mobile station in the multi-session system (resp. by the $i + j^{th}$ mobile station in the single-session system). RMS_i is the replicated part of the multi-session mobile station agent. Note that we group the name restrictions and we bring them in front of the process.

We define the grouped multi-session system component $GMS_{i,j}[\_]$ representing the leftovers after the execution of j sessions of the i^{th} mobile station and the simulating grouped single-session system component $GSS_{i,j}[\_]$ representing the leftovers after the execution of j single session mobile stations

simulating the j sessions of the i^{th} mobile station of the multi-session system, as follows:

$$GMS_{i,j}[\_] \stackrel{def}{=} \nu\, \widetilde{ms}_{i,j}.\nu\, \widetilde{nid}_{i,l}.(MMS^7_{i,1} \mid \cdots \mid MMS^7_{i,j-1} \mid \_ \mid !RMS_i)$$

$$GSS_{i,j}[\_] \stackrel{def}{=} \nu\, \widetilde{ss}_{i,j}.\nu\, \widetilde{nid}_{i,l}.(SMS^7_{i,1} \mid \cdots \mid SMS^7_{i,j-1} \mid \_) \quad l \in \{j-1, j\}$$

The grouped multi (resp. single)-session system components are the building blocks of the bisimulation relation. They basically define how the grouped single-session MSs can mimic the structure resulting by the evolution of a multi-session mobile station. We define the symmetric relation between the single-session and the multi-session system to be:

$$\mathcal{R} \stackrel{def}{=} \{(C,\ D),\ (D,\ C) : \exists\ n,\ m \geq 0,$$

$$A \equiv \nu\, dck.(C_1 \mid \cdots \mid C_n \mid PSN_m \mid !SSA \mid !SN),$$

$$B \equiv \nu\, dck.(D_1 \mid \cdots \mid D_n \mid PSN_m \mid !MSA \mid !SN),$$

where $\forall i,\ 1 \leq i \leq n$,
$\exists l_i,\ k_{l_i},\ l_i \geq 0, 1 \leq k_{l_i} \leq 8$ such that

$$C_i = GSS_{i,l_i}[SMS^{k_{l_i}}_{i,l_i} \mid SSN_{i,l_i}] = \nu\, \widetilde{ss}_{i,l_i}.\nu\, \widetilde{nid}_{i,j}.(SMS^7_{i,1} \mid \cdots \mid SMS^7_{i,l_i-1} \mid SMS^{k_{l_i}}_{i,l_i} \mid SSN_{i,l_i})$$

$$D_i = GMS_{i,l_i}[MMS^{k_{l_i}}_{i,l_i} \mid MSN_{i,l_i}] = \nu\, \widetilde{ms}_{i,l_i}.\nu\, \widetilde{nid}_{i,j}.(MMS^7_{i,1} \mid \cdots \mid MMS^7_{i,l_i-1} \mid MMS^{k_{l_i}}_{i,l_i} \mid MSN_{i,l_i} \mid !RMS_i)$$

$$SSN_{i,l_i} = MSN_{i,l_i} = SN^{h_1}_{i,1} \mid \cdots \mid SN^{h_{l_i-1}}_{i,l_i-1} \mid L^{h_{l_i}},\ h_1, \ldots, h_{l_i-1} \geq 2$$

$$L^{h_{l_i}} = \begin{cases} 0 & \text{if } k_{l_i} \in \{1,\ 2\} \\ SN^{h_{l_i}}_{i,l_i} & \text{otherwise} \end{cases}$$

$$j = \begin{cases} l_i - 1 & \text{if } L^{h_{l_i}} = 0 \\ l_i & \text{otherwise} \end{cases}$$

$$PSN_m = SN^1_{j_1} \mid \cdots \mid SN^1_{j_m},\ \text{for some } j_1, \ldots,\ j_m \in \{0,\ 1\}$$

$\}$

We want to prove that $\mathcal{R}$ is a bisimulation. To ease this proof, we define a lemma dealing with the bisimulation part of the proof and a lemma dealing with static equivalence. Informally, Lemma 1 states that the actions of a system can be mimicked by actions of the other system and vice versa. Formally:

Lemma 1: Let $C \equiv \nu\, dck.(C_1 \mid \cdots \mid C_n \mid PSN_m \mid !SA \mid !SN)$, $D \equiv \nu\, dck.(D_1 \mid \cdots \mid D_n \mid PSN_m \mid !\overline{SA} \mid !SN)$ such that $SA = SSA$ (resp. $SA = MSA$) and ($\overline{SA} = MSA$ (resp. $\overline{SA} = SSA$) and $(C, D) \in \mathcal{R}$
if $C \stackrel{\ell}{\rightarrow} C'$ with $fv(\ell) \subseteq dom(C)$ and $bn(\ell) \cap fn(D) = \emptyset$ then $D \stackrel{\ell}{\rightarrow} D'$ and $(C', D') \in \mathcal{R}$ for any $\ell \in \{\tau, \alpha\}$.

The proof of Lemma 1 relies on the proof of two extra lemmas which informally state that if the single (resp. multi)-session system can do a transition then either one of the grouped single (resp. multi)-session system components (i.e. one of the C_i, respectively D_i) can do the transition, possibly synchronizing with one of the SN^1_j components of the PSN_m process (i.e. the MS synchronizes with the SN. This step models the establishment of means for ciphering of the TMSI reallocation protocol); or one of the components under replication is unrolled and does the transition; or one of the mobile stations starts a new session (in case of the multi-session system). The details of the proof are available for inspection [29].

To complete the proof of Proposition 1 we have to prove that the processes obtained after each simulation step are statically equivalent. This is stated by Lemma 2.

Lemma 2: If $(C, D) \in \mathcal{R}$ then $C \approx_s D$

In order to ease this proof we define the following substitutions:

$$\sigma^{id}_{i,j} \stackrel{def}{=} \{{}^{id_{i,1}}/_{x_{i,1}}, {}^{id_{i,2}}/_{x_{i,2}}, \ldots, {}^{id_{i,j}}/_{x_{i,j}}\}$$

$$\sigma^{M}_{i,j} \stackrel{def}{=} \{{}^{id_{i,1}}/_{x_{i,1}}, {}^{M_{i,2}}/_{x_{i,2}}, \ldots, {}^{M_{i,j}}/_{x_{i,j}}\}$$

$$\sigma^{K}_{i,j} \stackrel{def}{=} \{{}^{\mathtt{senc}(ck_{i,1},\ mr_{i,1},\ \text{COMPLETE})}/_{k_{i,1}}, \ldots, {}^{\mathtt{senc}(ck_{i,j},\ mr_{i,j},\ \text{COMPLETE})}/_{k_{i,j}}\}$$

$$\sigma^{nid}_{i,j} \stackrel{def}{=} \{{}^{\mathtt{senc}(ck_{i,1},\ sr_{i,1},\ \mathtt{pair}(\text{TMSI\_REALL},\ nid_{i,1}))}/_{y_{i,1}}, \ldots, {}^{\mathtt{senc}(ck_{i,j},\ sr_{i,j},\ \mathtt{pair}(\text{TMSI\_REALL},\ nid_{i,j}))}/_{y_{i,j}}\}$$

Moreover, we prove in Lemma 3 that the structure of the frame of a single (resp. multi)-session system is as follows:

Lemma 3: Let $(C, D) \in \mathcal{R}, C \equiv \nu\, dck.(C_1 \mid \cdots \mid C_n \mid PSN_m \mid !SSA \mid !SN), D \equiv \nu\, dck.(D_1 \mid \cdots \mid D_n \mid PSN_m \mid !MSA \mid !SN)$ then $\varphi(C) \equiv \nu\, dck.(\varphi(C_1) \mid \cdots \mid \varphi(C_n))$, $\varphi(D) \equiv \nu\, dck.(\varphi(D_1) \mid \cdots \mid \varphi(D_n))$ where $\forall i,\ l_i\ 1 \leq i \leq n,\ l_i \geq 0$,

$$\begin{aligned} \varphi(C_i) &\equiv \varphi(GSS_{i,l_i}[SMS^{k_i}_{i,l_i} \mid SSN_{i,l_i}]) \\ &\equiv \nu\, \widetilde{ss}_{i,l_i}.\nu\, \widetilde{nid}_{i,j_{nid}}.(\sigma^{id}_{i,j_{id}} \mid \sigma^{K}_{i,j_K} \mid \sigma^{nid}_{i,j_{nid}}) \\ \varphi(D_i) &\equiv \varphi(GMS_{i,l_i}[MMS^{k_i}_{i,l_i} \mid MSN_{i,l_i}]) \\ &\equiv \nu\, \widetilde{ms}_{i,l_i}.\nu\, \widetilde{nid}_{i,j_{nid}}.(\sigma^{M}_{i,j_{id}} \mid \sigma^{K}_{i,j_K} \mid \sigma^{nid}_{i,j_{nid}}) \end{aligned}$$

We then use the obtained frame structure to define a ProVerif bi-process that generates the frame of the multi-session and single-session processes. This allows us to automatically prove the static equivalence. Hence, the full proof combines manual and automatic techniques. We can now easily prove that the TMSI reallocation procedure preserves unlinkability if a new session key is established before each execution by proving the following proposition.

Proposition 1: Let S and M be respectively the single and multi- session systems as defined in Section IV-B. We have that $S \approx_l M$.

Proof: We show that $(S,\ M) \in \mathcal{R}$.
Let $C = S$ and $D = M$, let $n = 0,\ m = 0$ then $C \equiv \nu\, dck.(!SSA \mid !SN) \equiv S$ and $D \equiv \nu\, dck.(!MSA \mid !SN) \equiv M$ and $(S,\ M) \in \mathcal{R}$.
We show that $\mathcal{R}$ is a bi-simulation.
We show that $C \approx_s D\ \forall\ (C, D) \in \mathcal{R}$: trivially follows by Lemma 2.
We show that if $C \stackrel{\tau}{\rightarrow} C'$ then $\exists\ \ D'$ such that $D \stackrel{D}{\Rightarrow}{}'$ and $(C',\ D') \in \mathcal{R}$: trivially follows by Lemma 1.

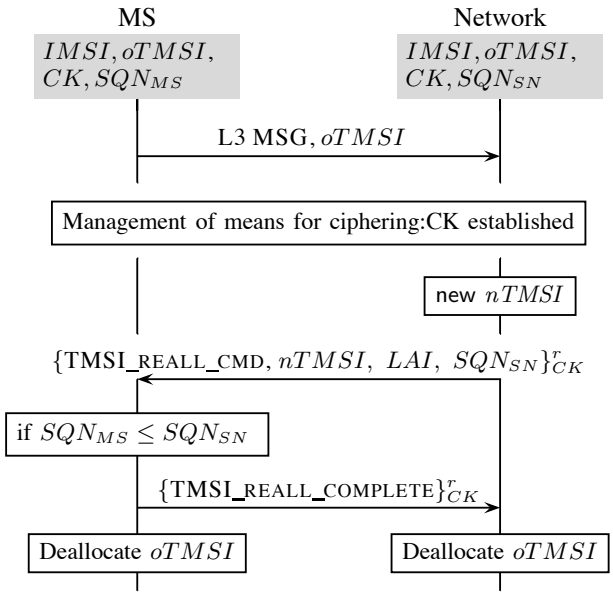

Fig. 8. TMSI Reallocation Procedure SQN Fix

We show that if $C \xrightarrow{\alpha} C'$ and $fv(\alpha) \subseteq dom(C), \quad bn(\alpha) \cap fn(D) = \emptyset$ then $\exists\ D'$ such that $D \Rightarrow\xrightarrow{\alpha}\Rightarrow D'$ and $(C',\ D') \in \mathcal{R}$: trivially follows by Lemma 1 ■

V. Discussion

The experiments we conducted show that the adoption of pseudonyms is not a sufficient condition to ensure the privacy of mobile telephony users and that real network implementations leave plenty of room for tracking attacks. We suggest network operators should adopt activity-related policies in order to prevent active tracking attacks. In general, the execution of the TMSI reallocation procedure should be more frequent even when the MS is in idle state, in order to prevent mere passive tracking.

Our formal analysis of the TMSI reallocation procedure considers a simplified version of the protocol and abstracts away the establishment of CK through the execution of the key agreement protocol. However, it allows us to show that the TMSI reallocation procedure should always be executed by first establishing new session keys, otherwise the TMSI reallocation does not guarantee the unlinkability property that it is meant to provide to the users and is then useless.

The solution we propose and formally verify does not require any change in the security architecture of mobile telephony systems. It only requires the standard to specify that the reuse of the encryption key is not permitted when the key is used to execute the TMSI reallocation procedure. However, frequent executions of the authentication procedure could burden the radio communication and slow down the delivery of mobile telephony services. Alternative solutions are possible, as for example the introduction of a sequence number in the TMSI reallocation command, similarly to the one used to avoid replay attacks against the Authentication and Key Agreement protocol [20]. We illustrate this solution in Figure 8. The network sends a sequence number SQN_{SN} along with the TMSI reallocation command. The MS checks if the received sequence number is in the expected range ($SQN_{MS} \leq SQN_{SN}$). If so it carries on with the reallocation of the TMSI. Otherwise the MS aborts the TMSI reallocation execution, hence avoiding replay attacks.

VI. Future Work

In this paper, we were concerned about misunderstanding regarding the capabilities of pseudonyms reallocation. Having identified critical scenarios in the real implementation of the pseudonym changing mechanism, and implementation details weakening the privacy of mobile telephony users making them linkable, we think it would be interesting to gather data about how widespread these issues are throughout the existing networks in an extensive and systematic way in order to calculate some interesting statistics. This for example would reveal if the critical scenarios are peculiar of a mobile network operator or instead are linked to some specific base station implementation and to estimate how widespread the user linkability problem is within mobile telephony systems.

As previously discussed, the TMSI reallocation procedure challenges the currently available state of the art tool for the automatic verification of cryptographic protocols. Firstly, because the encoding of internal states is needed in order to store the currently used MS identity and secondly because the modelling of privacy related properties requires automatic tools able to deal with the automatic verification of observational equivalence. We aim to address this issue in future by developing an extension of the StatVerif [28] tool capable of verifying observational equivalence properties.

VII. Conclusions

Using pseudonyms is a good mechanism to ensure the user's privacy, provided that there is enough possibility of mixing within the network (*i.e.* the user is not the only one in a given area) which is usually the case in mobile telecommunication networks. However, the efficiency of the pseudonym change strategy depends on many factors which the 3GPP standard leaves as implementation choices.

We showed that the implementation choices made by real network operators do not provide a satisfying level of privacy and leave space for different kinds of tracking attacks. Moreover, we showed that the standard specifications is flawed and the TMSI reallocation procedure is subject to a linkability attack when restored encryption keys are used.

Our analysis clarifies that the minimum criteria for the execution of the TMSI reallocation should be defined and mandated by the standard (otherwise users are linkable). These criteria should be activity, time and location dependent. Secondly, implementations that don't change TMSI at each change of location make tracking (even passive) easy and hence this should be forbidden by the standard. Finally, the establishment of new encryption keys before the execution of the TMSI reallocation should be compulsory (otherwise consecutively assigned TMSI are linkable).

The solution we propose as a countermeasure to the replay attack is easily and readily adoptable without changing the current system architecture, with the added value of having

formal guarantees on the achieved privacy properties. In fact, we formally proved that if new session keys are established for each TMSI reallocation execution then unlinkability is preserved. Our proof of unlinkability is one of the few examples in the literature of a proof of labelled bisimilarity of a real-sized protocol. Such manual proofs give useful insights on the way one could automate them, and thus pave the way to automating labelled bisimilarity proofs.

As future work we plan on confirming the replay attack experimentally, checking if there are or not mechanisms in place (not stated in the standard) to thwart this attack by preventing replayed messages from being accepted by the Mobile Station. Also, a thorough and methodical analysis of the level of privacy achieved by different privacy policies would be of great interest. However, this would possibly require collecting further data about user mobility, aggregation areas, population density, network coverage and user base per geographical area. This kind of analysis goes beyond the scope of the present work and is left as future work. Moreover, the impact of the adoption of the proposed policies on the network performances should be studied as well in order to balance the offered level of privacy accordingly.

ACKNOWLEDGMENT

We are thankful to EPSRC for supporting this work through the projects Verifying Interoperability Requirements in Pervasive Systems (EPSRC-reference EP/F033540/1) and Trust Domains (EPSRC-reference TS/I002529/1).

REFERENCES

[1] http://www.pathintelligence.com, path Intelligence Ltd. (2010) FootPath. [Online]. Available: http://www.pathintelligence.com

[2] http://www.smart-flows.com, smart Flows. [Online]. Available: http://www.smart-flows.com

[3] K. Rawlinson, "3G security flaw leaves smartphone users at risk of hackers ," *Independent*, October 2012.

[4] V. Cortier and B. Smyth, "Attacking and fixing helios: An analysis of ballot secrecy," *Journal of Computer Security*, 2012.

[5] V. Cortier and C. Wiedling, "A formal analysis of the norwegian e-voting protocol," in *Proceedings of the 1st International Conference on Principles of Security and Trust (POST'12)*, ser. Lecture Notes in Computer Science, vol. 7215. Tallinn, Estonia: Springer, Mar. 2012, pp. 109–128.

[6] M. Barbaro and T. Zeller Jr., "A face is exposed for AOL searcher no. 4417749," The New York Times, August 9, 2006.

[7] C. Caldwell, "A pass on privacy?" The New York Times, July 17, 2005.

[8] D. Goodin, "Defects in e-passports allow real-time tracking," the Register, 26th January 2010.

[9] G. Avoine and P. Oechslin, "RFID Traceability: A Multilayer Problem," in *Financial Cryptography*, ser. FC, 2005.

[10] A. Biryukov, A. Shamir, and D. Wagner, "Real time cryptanalysis of A5/1 on a PC," in *Proceedings of the 7th International Workshop on Fast Software Encryption*, 2000, p. 118.

[11] K. Nohl and S. Munaut, "Wideband GSM sniffing," http://events.ccc.de/congress/2010/Fahrplan/attachments/1783_101228.27C3.GSM-Sniffing.Nohl_Munaut.pdf.

[12] Z. Ahmadian, S. Salimi, and A. Salahi, "New attacks on UMTS network access," in *Conference on Wireless Telecommunications Symposium*, ser. WTS'09, 2009.

[13] D. F. Kune, J. Koelndorfer, N. Hopper, and Y. Kim, "Location leaks over the GSM air interface," in *Annual Network & Distributed System Security Symposium*, ser. NDSS, 2012.

[14] H. Welte, S. Munaut, A. Eversberg, and other contributors, "OsmocomBB," http://bb.osmocom.org.

[15] T. Engel, "Locating mobile phones using signalling system 7," http://events.ccc.de/congress/2008/Fahrplan/attachments/1262_25c3-locating-mobile-phones.pdf, 25th Chaos Communication Congress (25C3).

[16] K. Nohl and L. Melette, "Defending mobile phones," http://events.ccc.de/congress/2011/Fahrplan/attachments/1994_111217.SRLabs-28C3-Defending_mobile_phones.pdf, 28th Chaos Communication Congress (28C3). [Online]. Available: http://www.gsmmap.org

[17] S. May and L. Melette, "Distributed GSM security analysis," https://program.sigint.ccc.de/fahrplan/system/attachments/21/original/120518.GSMMAP-SIGINT.pdf, sIGINT 2012. [Online]. Available: http://www.gsmmap.org

[18] M. Arapinis, L. I. Mancini, E. Ritter, M. Ryan, N. Golde, K. Redon, and R. Borgaonkar, "New privacy issues in mobile telephony: fix and verification," in *ACM Conference on Computer and Communications Security*, 2012, pp. 205–216.

[19] 3GPP, "Technical specification group core network and terminals; mobile radio interface layer 3 specification; core network protocols; stage 3 (release 9), TS 24.008," 2010.

[20] ——, "Technical specification group services and system aspects; 3G security; security architecture (release 9), TS 33.102," 2010.

[21] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi, "Understanding individual human mobility patterns," *Nature*, vol. 453, pp. 779–782, 2008. [Online]. Available: http://www.nature.com/nature/journal/v453/n7196/suppinfo/nature06958_S1.html

[22] M. Bayir, M. Demirbas, and N. Eagle, "Discovering spatiotemporal mobility profiles of cellphone users," in *World of Wireless, Mobile and Multimedia Networks Workshops, 2009. WoWMoM 2009*, june 2009, pp. 1–9.

[23] G. Combs, "Wireshark," http://www.wireshark.org.

[24] "traces," http://markryan.eu/research/mobile/tmsi-reallocation/.

[25] M. Arapinis, T. Chothia, E. Ritter, and M. Ryan, "Analysing unlinkability and anonymity using the applied pi calculus," in *IEEE Computer Security Foundations Symposium*, ser. CSF, 2010.

[26] M. Abadi and C. Fournet, "Mobile values, new names, and secure communication," in *ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, ser. POPL, 2001.

[27] B. Blanchet, "Proverif: Cryptographic protocol verifier in the formal model," http://www.proverif.ens.fr/.

[28] M. Arapinis, E. Ritter, and M. Ryan, "Statverif: Verification of stateful processes," in *IEEE Computer Security Foundations Symposium*, ser. CSF, 2011.

[29] "proof," http://markryan.eu/research/mobile/tmsi-reallocation/proof.pdf.